# Passengers Survival Prediction

# Using Ensemble Machine Learning Model

**Your Name Here**

**Your ID**

Contents

[Passengers Survival Prediction 1](#_Toc100763468)

[Using Ensemble Machine Learning Model 1](#_Toc100763469)

[Introduction: 3](#_Toc100763470)

[Background: 3](#_Toc100763471)

[Literature Review: 3](#_Toc100763472)

[Experimental design: 4](#_Toc100763473)

[Data Pre-Processing: 5](#_Toc100763474)

[Outlier detection: 5](#_Toc100763475)

[Checking for missing values: 6](#_Toc100763476)

[Feature Analysis: 7](#_Toc100763477)

[Model Building: 10](#_Toc100763478)

[Splitting the data: 11](#_Toc100763479)

[Simple Model Building: 11](#_Toc100763480)

[Cross-Validation: 11](#_Toc100763481)

[Hyperparameter tuning for best models: 12](#_Toc100763482)

[Ensemble modeling: 15](#_Toc100763483)

[Combine the Models: 16](#_Toc100763484)

[Conclusion: 17](#_Toc100763485)

[Bibliography 18](#_Toc100763486)

# Introduction:

Outfit learning or ensemble learning is an umbrella term for techniques that consolidate numerous inducers to go with a choice or decision, normally in supervised machine assignments. An inducer likewise alluded to as a base-student, is a calculation that takes a bunch of named models as info and produces a model (e.g., a classifier or regressor) that sums up these models. By utilizing the created model, forecasts can be drawn for new unlabeled models. A troupe inducer can be of an AI calculation (machine learning) (e.g., decision tree, neural network, linear regression model, and so on.). The principal reason for ensemble learning is that by joining different models, the blunders of a solitary inducer will probably be repaid by different inducers, and subsequently, the general forecast execution of the group would be preferable over that of a solitary inducer.

# Background:

Ensemble Methods are among the most impressive and least demanding for utilization of prescient investigation calculations and R programming language and Python has an extraordinary assortment that incorporates the best entertainers - Random Forest, Gradient Boosting, and Bagging as well as large information forms that are accessible through Revolution Analytics.

The expression "ensemble Methods" for the most part alludes to building countless fairly free prescient models and afterward consolidating them by casting a ballot or averaging to yield extremely elite execution. Troupe techniques have been called publicly supporting machines. Stowing, Boosting and Random Forest all have the goal of further developing execution past what's attainable with a parallel choice tree, however, the calculations adopt various strategies to further develop execution.

Maybe one of the earliest works on ensemble learning is Dasarathy and Sheela's 1979 paper (Dasarathy 1979), which previously proposed involving an ensemble system in separation and overcome style, parceling the element space utilizing at least two classifiers. More than 10 years after the fact, Hansen and Salamon (Hansen 1990) showed the change decreased property of ensemble learning, and that the speculation execution of a brain organization can be improved by utilizing a troupe of also designed neural networks.

# Literature Review:

Ensemble learning is Group learning which is normally viewed as the ML translation for the insight of the group. This idea can be represented through the tale of Sir Francis Galton (1822-1911) who was an English thinker and analyst that imagined the essential idea of standard deviation and relationship. While visiting an animal’s fair, Galton led a basic weight speculating challenge. The members were approached to figure out the heaviness of a bull. Many individuals took part in this challenge, yet nobody prevailed with regards to speculating the weight: 1,198 pounds. Causing him a deep sense of shock, Galton found that the normal of all surmises came very near the specific weight: 1,198 pounds. In this trial, Galton uncovered the force of joining numerous expectations to get a precise forecast. Troupe strategies manifest this idea in AI challenges, where they bring about superior prescient execution contrasted with a solitary model. Likewise, when the computational expense of the partaking inducers is low (e.g., choice tree), outfit models are frequently extremely productive [1].

Ensemble learning is also helpful and can be used in forecasting which is a displaying approach that joins data sources, and models of various kinds, with elective presumptions, utilizing particular example acknowledgment techniques. The point is to involve all suitable data in expectations, without the restricting and inconsistent decisions and conditions coming about because of a solitary factual or AI approach or a solitary practical structure, or results from a restricted information source. Vulnerabilities are methodically represented. Results of outfit models can be introduced as a scope of conceivable outcomes, to show how much vulnerability is displayed [2].

The noteworthy adaptability and flexibility of ensemble techniques and deep learning models have prompted the multiplication of their application in bioinformatics research. Customarily, these two AI strategies have generally been treated as autonomous philosophies in bioinformatics applications. In any case, the new development of gathering deep learning-wherein the two machine learning strategies are joined to accomplish synergistic upgrades in model exactness, steadiness and reproducibility-has provoked another rush of exploration and application. The ensemble learning techniques are also useful not only in machine learning but also in forecasting and deep learning [3].

From the literature study, we came to know that ensemble learning is not only used in regression and classification using machine learning, but it is also helpful in time series forecasting and deep learning neural networks.

# Experimental design:

The goal of this project is to use ensemble learning techniques to perform the classification of the data and perform predictions. For this project we took the dataset of Titanic from here <https://data.world/nrippner/titanic-disaster-dataset>

The goal is to predict the survival of the passengers, the feature Survival has binary values like YES and NO, Yes indicates that the passenger survived and No means the passenger didn’t survive. When we are about to predict binary data then we use the classification, ensemble learning helps us to perform binary classification also.

The goal is to perform the data preprocessing and after doing all the cleaning and data transformation, we will use different ensemble methods to predict the survival of the passengers, we will perform the hyperparameter tuning and then combine the best models we get to the base model and perform the prediction.

## Data Pre-Processing:

Data pre-processing is the process where we apply different techniques to clean the data and transform data to prepare it for machine learning. In real-world datasets, there is a huge possibility that the data is not in a good shape, we may have lots of missing values, outliers, and categorical features, and we have to clean the data for machine learning.

In our dataset, we have features like Passenger ID, their names, the fares they paid, their Cabins, etc, and the Survival feature which shows whether the passenger survived the incident or not. The step-by-step process we did is given below.

* Outlier detection
* Checking for Missing values
* Feature Analysis
* Categorical Features
* Modeling

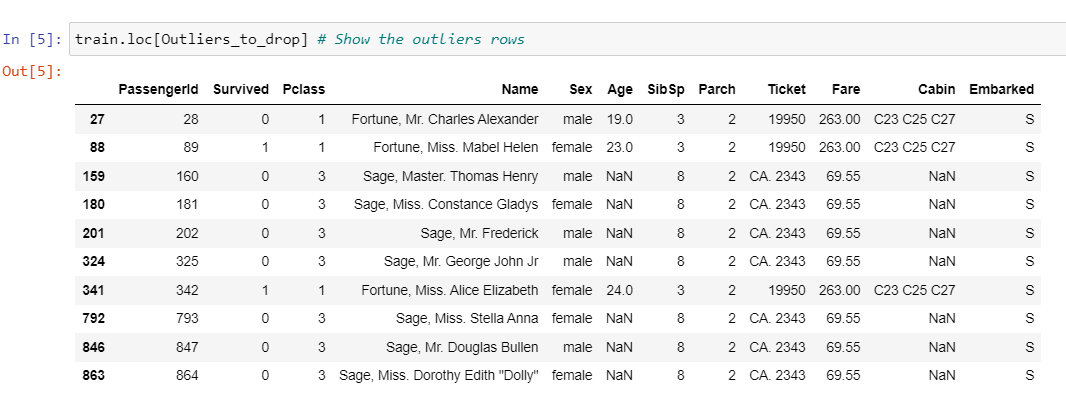
We’ll explain the above in-depth.

### Outlier detection:

An outlier is a data point that behaves differently from the rest of the data, for example, a value that lies much smaller or much bigger than the rest of the data values in the dataset. The outliers increase the variability in a dataset, which decreases the statistical power of the data, removing them helps us get the best results.

There are many ways to handle outliers, some are by handling them using the standard deviation methods and IQR method. Sometimes we remove them.

Upon checking the outliers in our dataset, we got some of them, the outliers in our dataset is shown in the following table.



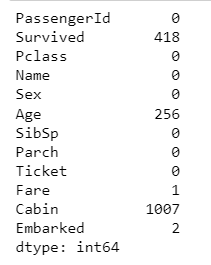
The above table shows the outliers in our dataset

We detect 10 outliers. The 28, 89, and 342 passengers have a high Ticket Fare The 7 others have very high values of SibSP. Now we are dropping the outliers from our dataset by using the Pandas (drop()) function.

### Checking for missing values:

Missing values are the main problem in any dataset, these are those values that are not recorded at the time of data collection, or these values may be deleted or lost from the dataset. Handling the missing values is one of the most important steps that a data analyst and scientist should take. We handle the missing values using different techniques, these techniques include some statistical methods like Mean, Median, and Mode, we use these methods to fill the missing values and sometimes we remove them.

We checked the missing values in our dataset and got the following results.



Missing values

We have missing values in the data, as you can see in the above table, we have 418 missing values in the Survived feature and 256 in Age, similarly 1007 in the Cabin and only 2 in the Embarked feature. We will simply replace the NaN values with nan.

### Feature Analysis:

Feature Extraction is the process to take all the useful features from the dataset which are useful for machine learning predictions. We removed all the other features except Survived, SibSp, Parch, Age, Fare, we plotted a correlation plot of these taken features to see their correlation with the Survived feature.



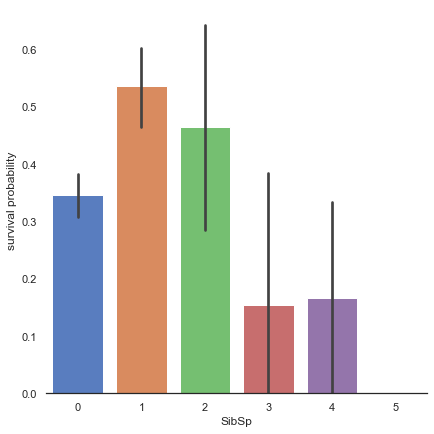
Correlation plot

The correlation plot showed the Fare feature which is a significant correlation with the survival probability. It doesn't mean that the other features are not useful. Subpopulations in these features can be correlated with survival. To determine this, we need to explore in detail these features.

Now let’s check the other features and do some analysis.

##### SibSP:

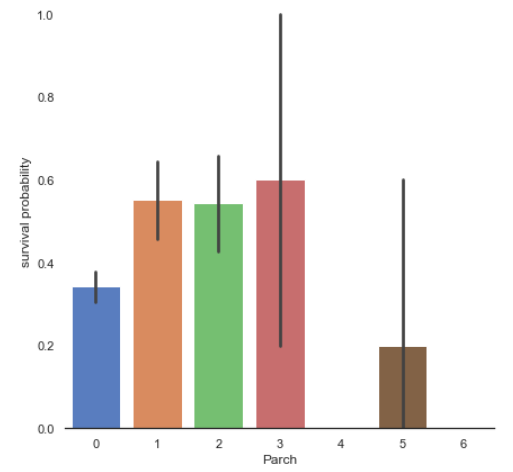
Let’s explore the SibSP feature concerning Survived feature.



It seems that passengers having a lot of siblings/spouses have less chance to survive, single passengers (0 SibSP) or with two other persons (SibSP 1 or 2) have more chance to survive. This observation is quite interesting.

#### Parch:

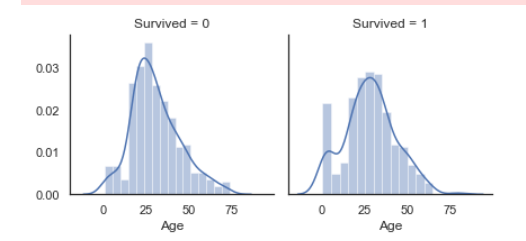
Explore Parch feature vs Survived.



Small families have more chances to survive, more than single (Parch 0), medium (Parch 3,4), and large families (Parch 5,6).

#### Age:

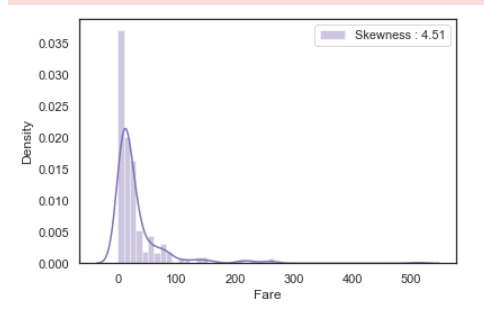
Explore the Age feature VS the Survived feature.



Age distribution seems to be a tailed distribution, maybe a gaussian distribution. We notice that age distributions are not the same in the survived and not survived subpopulations. Indeed, there is a peak corresponding to young passengers, that have survived. We also see that passengers between 60 and 80 have survived. So, even if "Age" is not correlated with "Survived", we can see that there are age categories of passengers that have more or less a chance to survive. It seems that very young passengers have more chance to survive.

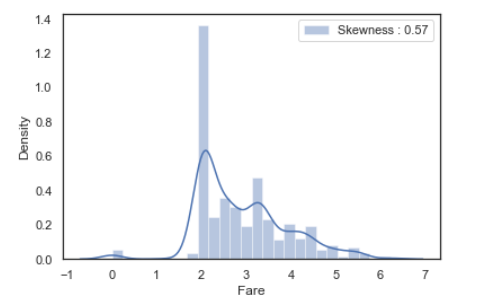
#### Fares:

Exploring Fares column.



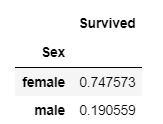
As we can see, Fare distribution is very skewed. This can lead to overweight very high values in the model, even if it is scaled. In this case, it is better to transform it with the log function to reduce this skew.

After applying log to Fare to reduce skewness distribution, we got the following results.



Skewness is reduced after the log transformation.

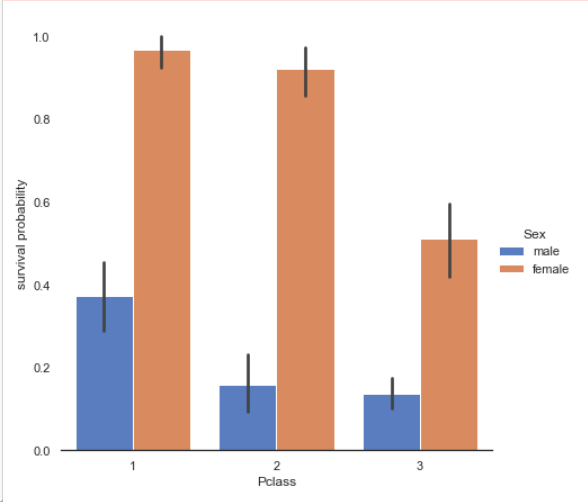
#### Gender:



Males have less chance to survive than females. So, Sex might play an important role in the prediction of survival. For those who have seen the Titanic movie (1997), I am sure, we all remember this sentence during the evacuation: "Women and children first".

#### PClass:

We got some useful information while analyzing the PClass feature.



Passenger survival is not the same in the 3 classes. First-class passengers have more chances to survive than second-class and third-class passengers. This trend is conserved when we look at both male and female passengers.

# Model Building:

Now after performing all the analysis and advanced EDA of the data, the next step is to build a model to perform predictions on the data. The goal is to predict the Passenger Survival probability based on all the other useful features. We are going to use some ensemble models like Gradient Boosting, SVM, KNN, and Random Forest, we will first train these models individually and do the predictions then we will use a base model and use all these four models together to predict.

We are taking the following steps in the model building.

* Splitting the data
* Simple modeling
* Ensemble modeling

## Splitting the data:

Splitting the data into pieces is a common practice in machine learning, this way we define the training data and the testing data. On the training data, we train the machine learning model, and on the testing data, we test the model and perform predictions on the model using testing data.

In our case, we have two datasets, one dataset is the training dataset and the other one is the test dataset. The training dataset is the one we analyzed throughout the exploratory data analysis and the testing dataset will be used to test our models. After doing so, we will step into the model building.

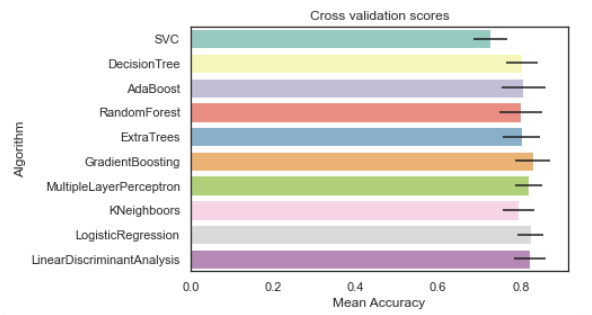
## Simple Model Building:

We are going to train different models, the models that we are training are given below. The goal is to first train these models individually, after that we will use all of them together to perform the predictions. We will first use the cross-validation process to see the performance of the models, based on the performance, we will select those models for ensemble learning.

* SVC
* Decision Tree
* AdaBoost
* Random Forest
* Extra Trees
* Gradient Boosting
* Multiple layer perceptron (neural network)
* KNN
* Logistic regression
* Linear Discriminant Analysis

### Cross-Validation:

Cross-validation is the process to evaluate different models on the dataset to see the performance of all the models. Based on the performance we will select our models for ensemble learning.



Cross-validation score of the models

After performing the cross-validation process, we got the above results, now as there is a restriction to using some new models in the ensemble learning so we have selected the following models.

* SVC
* Random Forest
* Gradient Boosting
* KNN

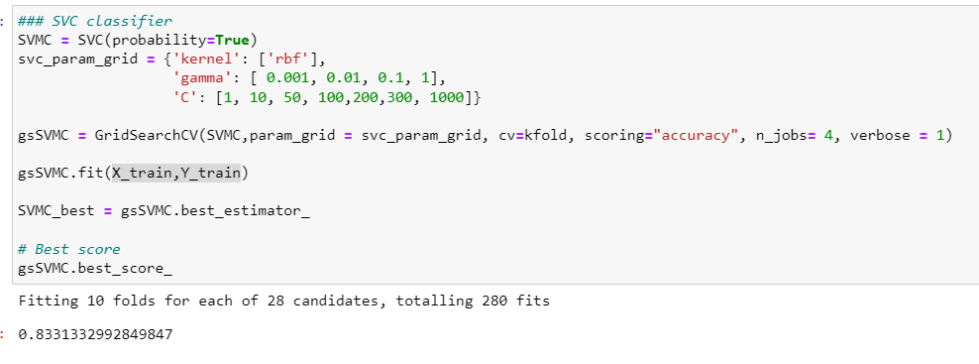
We first trained the above models individually, then we performed the parameter tuning of each model, then we combine all of them into one base model for ensemble learning.

### Hyperparameter tuning for best models:

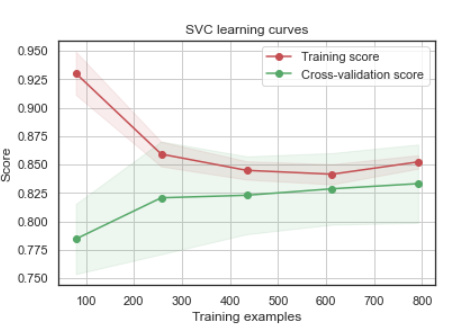
Hyperparameter tuning is the technique where we train the model on different parameters to see if the model gives good accuracy on any specific parameter? For this purpose, we used the Grid Search technique which helps us to search for the best parameters automatically.

#### SVC:

For SVC, we performed the Grid Search, the search gave us the best parameters, after training the SVC on those parameters, we got an accuracy of 83%.



Grid search parameter tuning of SVC



Learning Curve of SVC

We are passing the parameters of SVC to the GridSearchCV, then training the model on the best parameters, we got an accuracy of 83%.

#### Gradient Boosting:

The same technique will be followed for Gradient Boosting too.



Grid search parameter tuning of Gradient Boosting



The learning curve of gradient boosting

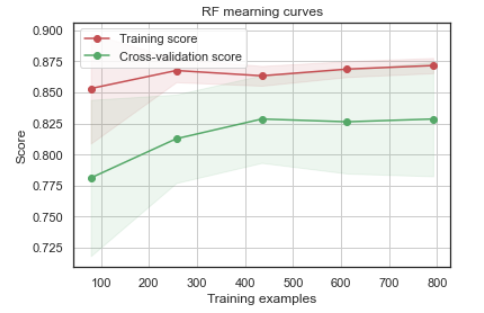
We got the accuracy of 82% on gradient boosting after training it on the best parameters mentioned by the GridSearchCV.

#### Random Forest:

The random forest also gave us an accuracy of 83%.



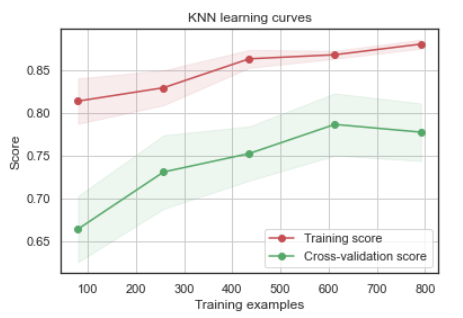
Grid search parameter tuning of RF



Learning Curve of Random Forest

#### KNN:

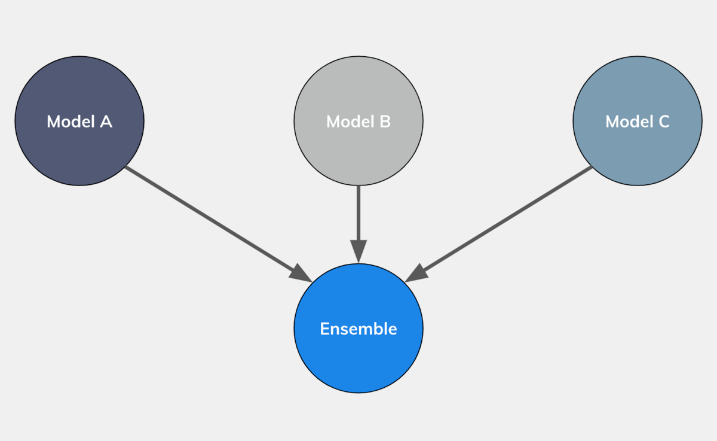
We trained the KNeighbor classifier and got the following learning curve



Learning curve of KNN

## Ensemble modeling:

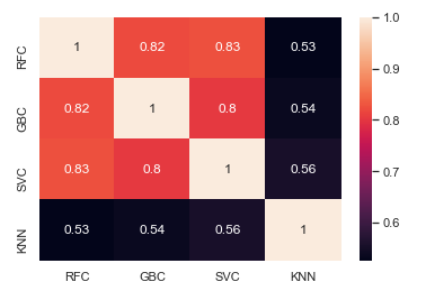
Ensemble Learning helps improve machine learning results by combining several models to improve predictive performance compared to a single model. The goal is to now combine all the above models that we have trained together into one base model, this is called ensemble modeling or learning.



Ensemble learning demonstration

The above graphs show the easy demonstration of ensemble learning, we combine different machine learning models into one base model.

We draw the correlation plot of the models that we have trained.

.

Results correlation of the trained models

The prediction seems to be quite similar. The 4 classifiers give more or less the same prediction but there are some differences. These differences between the 5 classifier predictions are sufficient to consider an ensemble vote.

### Combine the Models:

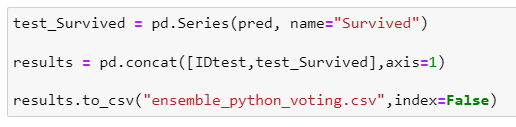
We choose a voting classifier to combine the predictions coming from the 5 classifiers. we preferred to pass the argument "soft" to the voting parameter to take into account the probability of each vote.

A Voting Classifier is a machine learning model that trains on an ensemble of numerous models and predicts an output (class) based on the highest probability of chosen class as the output. It simply aggregates the findings of each classifier passed into Voting Classifier and predicts the output class based on the highest majority of voting. The idea is instead of creating separate dedicated models and finding the accuracy for each of them, we create a single model which trains by these models and predicts output based on their combined majority of voting for each output class.



We passed all the above-trained models into the base model which is Voting Classifier, then we trained the base classifier again on the data, then we performed the prediction.

We are storing the predictions into a separate CSV file by using the following script.



Storing predictions into CSV file

# Conclusion:

The goal of this project is familiarized with the ensemble learning in machine learning, with the help of this method, we can put together different machine learning models into one base model. We started with the data pre processing by looking at each feature in depth and find some useful insights from them. We found outliers in our dataset; we removed all the outliers. Found some missing values. Then we choose some ensemble model and trained them individually, after that we selected the best models from a base ensemble classifier. We trained the ensemble model and saved the predictions into a CSV file.

The overall journey was so informative, we got to learn many new things from the data.

# Bibliography

|  |  |
| --- | --- |
| [1] | L. R. Omer Sagi, "Ensemble learning: A survey," *Wiley,* no. 2018, 2017. |
| [2] | D. L. Hao Wu, "The ensemble approach to forecasting: A review and synthesis," *Transportation Research Part C: Emerging Technologies,* 2021. |
| [3] | T. A. G. J. Y. H. Y. &. P. Y. Yue Cao, "Ensemble deep learning in bioinformatics," *Nature Machine Intelligence,* 2020. |